Citation/Reference	Chao Shang, Xiaolin Huang, Johan A.K. Suykens, Dexian Huang (2015) **Enhancing dynamic soft sensors based on DPLS : a temporal smoothness regularization approach** Journal of Process Control, vol. 28, pp. 17-26.
Archived version	Author manuscript : the content is identical to the content of the published paper, but without the final typesetting by the publisher
Published version	http://dx.doi.org/10.1016/j.jprocont.2015.02.006
Journal homepage	http://www.elsevier.com/locate/jprocont
Author contact	your email xiaolin.huang@esat.kuleuven.be +32/16/32 74 67 Klik hier als u tekst wilt invoeren.
Abstract	
IR	url in Lirias https://lirias.kuleuven.be/handle/123456789/4886666

(article begins on next page)

Enhancing Dynamic Soft Sensors based on DPLS: a Temporal Smoothness Regularization Approach

Chao Shang[a], Xiaolin Huang[b], Johan A.K. Suykens[b], Dexian Huang[a,*]

[a] *Tsinghua National Laboratory for Information Science and Technology (TNList) and Department of Automation, Tsinghua University, Beijing, P.R. China*
[b] *Department of Electrical Engineering (ESAT-STADIUS), KU Leuven, B-3001 Leuven, Belgium*

Abstract

Without an inclusion of process dynamics, traditional data-driven soft sensors are termed as static because only single snapshots of process samples are used. It leads to a series of limitations, such as sensitivity to temporal noises and inaccurate description in process dynamics. To this end, static models have been extended to dynamic versions thereof like dynamic partial least squares (DPLS) with lagged inputs for the sake of process dynamics. The dimension of soft sensor models inputs, however, could be considerably larger than static ones, which leads to the over-fitting problem. In this paper, we introduce the concept of temporal smoothness as a novel approach to DPLS-based dynamic soft sensor modeling. The starting point is to not only include historical process data but also impose smoothness regularization on proximal dynamic parameters. The smoothness regularization assumes that historical inputs have smoothly varying impacts on the latent variables as a valid prior knowledge, which is in consensus with the physical truth of industrial processes. Abrupt changes in model dynamics are desirably penalized and soft sensors therefore enjoy better generalizations and interpretations. A numerical example is given to demonstrate the advantages of temporal smoothness. A simulated Tennessee Eastman process study and a real quality prediction task in a crude distillation unit process are

*Corresponding author
Email address: huangdx@tsinghua.edu.cn (Dexian Huang)

provided to show the feasibility as well as effectiveness of our method.

Keywords: dynamic soft sensor, quality prediction, process control, dynamic PLS, temporal smoothness regularization

1. Introduction

In industrial processes, product quality is a paramount focus because of its close relationship with economic interests. To ensure operational safety and improve economic benefits, process practitioners have devised a wide variety of quality-relevant control and optimization schemes to assist process operations [1], [2], [3]. Because measurements of some quality indices are unavailable or costly, the soft sensing technique has been extensively researched and carried out in the process control community due to many advantages. It can provide a real-time and reliable source of product quality estimations in a cost efficient way. Traditionally, the most-used soft sensors are data-driven ones, which are constructed on the basis of massive data archived by distributed control systems (DCS) and are less dependent on specific process knowledge [4]. With the rapid development of information technology and computer sciences, applications of statistical inference and machine learning methods have been a major trend in modeling data-driven soft sensors. The most representative examples in this field are partial least squares (PLS) [5], [6], [7], artificial neural networks (ANN) [8], [9] and support vector machine (SVM) [10], [11].

Irregular and non-uniform sampling is a critical characteristic of quality variables in industrial processes [12], as described in Fig. 1. Quality samples are attained manually via laboratory analysis, which commonly takes a long time to accomplish. Consequently, quality samples are unavailable in most of the time, as described by red crosses in Fig. 1. The sampling interval for quality variables is extremely long, sometimes even longer than the process settling time. Such a long sampling interval renders successive quality samples almost statistically independent. In the context of traditional soft sensors, however, process data usually have to be down sampled to synchronize with quality data at an irreg-

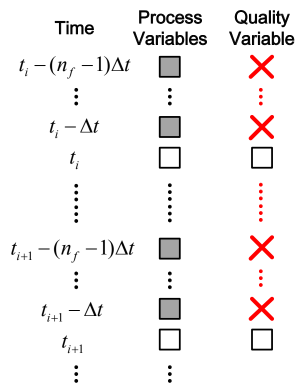

Figure 1: Irregular sampling of quality variables in chemical processes. t_i: the sampling time of quality data. Δt: the sampling interval of process data.

ular moment t_i, described as white boxes in Fig. 1. It is noticed that most process samples are overlooked by traditional soft sensor models, as denoted by grey boxes. Only a small portion of process data in seperate snapshots are adopted, under the assumption that processes work in static states, and quality variables are entirely dependent on the process variables sampled at the same snapshot. In this sense, traditional soft sensors are deemed as static. However, there are undoubtfully significant dynamics in industrial processes. The quality data are herein relevant with not only current process data, but also historical data in a period of time. The static assumption makes traditional soft sensors inadequate in description of process dynamics, because only a single snapshot of current moment is used while other informative historical data (grey boxes) are unfortunately ignored. It is inevitable that the estimation accuracy of static soft sensors would degrade easily with the presence of evident process dynamics.

Aside from above concerns in estimation inaccuracy, static soft sensors have some fundamental limitations in practical use. Fig. 2 displays a schematic structure of quality control loops in industrial processes. Because of the scarcity and large time delay involved in the quality measurements y, they cannot meet the demand of quality control. Instead, soft sensors provide real-time and continuous estimations $\hat{y}$ for the quality index, which serve as negative feedback signals

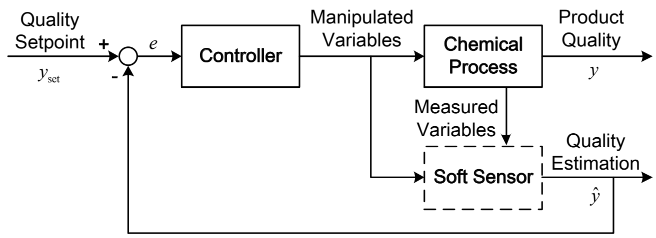

Figure 2: Sketch of a canonical quality control loop.

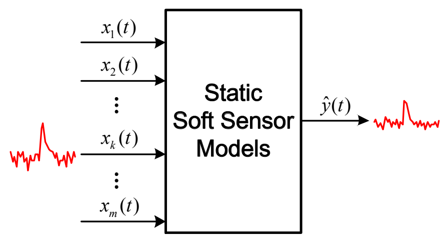

Figure 3: Sketch of a traditional static soft sensor and its instant response to short-term flutuations.

for quality control loops. The controller then automatically adjusts manipulated variables to maintain product quality according to the difference between the setpoint value and the real-time estimation from soft sensors. Therefore, prediction accuracy is especially crucial for control loops to work in a smooth and stable condition. Specifically, soft sensors should not only work well in regular conditions, but also provide reliable estimations under the circumstance of process uncertainties, such as short-term noises and fluctuations. Unfortunately, with only separate snapshots adopted, static soft sensors are sensitive to short-term fluctuations, which extensively exist in industrial processes. Ref. [13] revealed that short-term noises in predictions of a steady-state model could be harmful to control loop performances. As shown in Fig. 3, a short-term fluctuation in one process variable $x_k(t)$, for example, measurement noises or active adjustments of the controller, would impel the model prediction $\hat{y}(t)$ to respond immediately. The product quality, however, is more likely to have slow and smooth variations in practical scenarios. This is because the process itself contains evident dynamics, and short term fluctuations in process variables

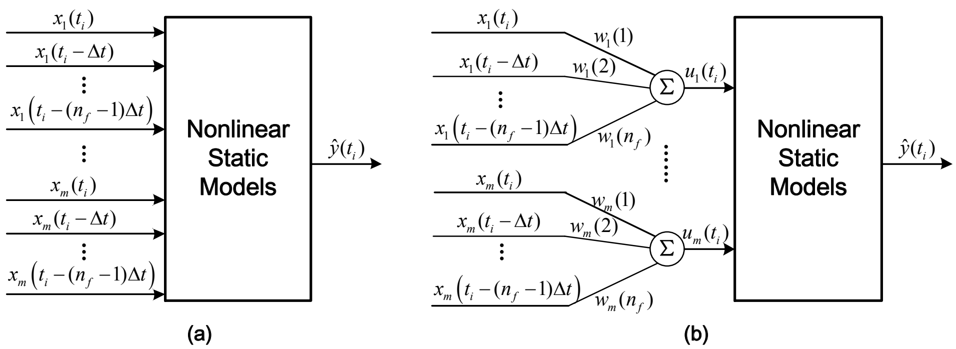

Figure 4: Sketches of SA & FF models. (a): SA model; (b): FF model. n_f: the length of historical data vectors.

ought to affect the product quality *gradually* rather than *instantaneously*. In this sense, static soft sensors declare no dynamics and fail to track long-term variations in processes, thereby being invalid surrogate models for inferential control. In practical scenarios, it is ubiquitous that process practitioners not only filter process data to smoothen individual noises before feeding them to soft sensors, but also filter online quality estmations to avoid abrupt variations. In this way, the dynamic information could be to some extent fused into the static model. Nevertheless, the design of such filters are excessively casual, mainly because it necessitates in-depth knowledge about a specific process, such as the causality relationship between variables and response times of mutual interactions, which are rather hard-won in practice.

To deal with the weakness of static soft sensor models in long-term smoothness, researchers have proposed different dynamic extensions by taking into consideration the time series data in a historical period rather than in a single snapshot. Such models could extract dynamic information from time series historical data, and can effectively filter the input noise to achieve a smooth estimation. In general, current dynamic soft sensor models can be classified into two groups, named as the simple augmented (SA) models and the FIR-filtered (FF) models respectively [14]. Fig. 4 depicts the structures of both SA and FF models. The SA model simply encompasses more lagged data as inputs of ordinary nonlinear static models. Bhat et al. first extended neural networks with historical samples to predict pH values in a continuous stirred-tank reactor (CSTR) [15]. A major pitfall of SA model is that its input dimensions would increase by n_f times if lagged data with length n_f are included. Different from the SA model, FF model has a clear physical interpretation. It postulates that the prediction model can be well approximated by a Wiener structure. The linear dynamic part is usually shaped as a first-order or second-order finite impulse response (FIR) model $\{w_k(l)\}$ $(1 \leq l \leq n_f)$ to capture dynamics, whereas the nonlinear part is responsible for the description of steady nonlinearities [16], [17]. An intermediate variable $u_k(t_i)$ as a dynamic feature is obtained by weighing lagged data of each variable with a linear FIR and thereafter acts as the input of ordinary nonlinear static models [18]. A large variety of models have been adopted as the nonlinear part in FF models, such as ANN [19], SVM [20]. Note that the input dimension of the nonlinear part in FF model remains equivalent to that in traditional static models. As a consequence, the over-fitting problem induced in the nonlinear block, is to some extent mitigated in contrast to SA models. Nevertheless, the optimization problems pertaining to FF models are usually non-convex, being rather strenuous to disentangle. The main focus of this article is hence on the SA models.

In the scale of SA dynamic models, PLS has been first modified to its dynamic extension (DPLS) due to its popularity in chemometrics. To improve the performance of inferential control, Kano [13] established a prediction model using DPLS for product composition in a distillation column, and found that the control performance was greatly improved. Lin et. al [21] proposed a systematic framework for soft sensor development, and used DPLS to address the auto-correlation in time series historical data. Galicia et. al [22] further proposed RO-DPLS to reduce the less relevant historical data in virtue of prior process knowledge. Although process dynamics could be addressed by extension with time-lagged process variables, the dimension of model inputs grows dramatically. In general, the more input variables a model has, the more complicated the model becomes, and the more training samples we need to guarantee a satisfactory generalization. Unfortunately, quality data are always sampled at a low frequency in practical scenarios, leading to a limited number of available data. Consequently, the directly augmented model is inevitably prone to the over-fitting problem when the sample size is small. Helland [23] explained the fact that PLS is not an optimal regression model with an excess of input variables. In such situations, the regularization acts as a prevailing statistical solution to the over-fitting straits. It serves to penalize extreme complexity in models and yields a simple structure in spite of massive model parameters. The most representative one is the L_2 norm regularization employed in SVM [24] and LS-SVM [25] to shrink model parameters towards zero. In addition, we can effectively integrate prior knowledge into data-based models via the regularization approach, making models more interpretable. For instance, the L_1 norm regularization can be used to induce sparse models in a wide range of tasks like compressive sensing [26], where sparsity is of special interest a priori. The regularization strategy herein has the potential to circumvent the over-fitting barrier to enhance the prediction accuracy as well as long-term smoothness of SA models. In the extensive literature of dynamic soft sensor modeling, however, regularization strategies have not received enough attentions ever since.

In this article, we propose a SA soft sensor model with temporal smoothness regularization by reformulating the optimization problem of DPLS, denoted as DPLS-TS. This formulation comprises a prior knowledge in sequential relationships of fast-sampled process data, which penalizes significant changes in impacts of proximal historical inputs on the model. In this way the soft sensor becomes interpretable with dynamic smoothness, being able to filter the short-term noise and capture the long-term trend. The formulation can be further cast as an eigenvector problem similar to the ordinary PLS, which provide computational convenience for practical usage. We show that the proposed model enjoys advantages in terms of dynamic interpretability and prediction accuracy.

This article is organized as follows: Section 2 clarifies notations and reviews the ordinary PLS algorithm. In Section 3, we propose the DPLS model with temporal smoothness by exploiting the relationship between dynamic inputs and the latent structure, and then establish the design procedure of soft sensors. Section 4 presents two simulated cases to illustrate the feasibility as well as effectiveness of the proposed model. In Section 5, a industrial application case study is provided to demonstrate the improvement of the proposed method contrary to DPLS, followed by concluding remarks and future work comments in Section 6.

2. Preliminaries

2.1. Notations in SA models

Before reviewing the ordinary PLS approach, some notations including lagged variables and model coefficients, are clarified in this subsection. Assume that there are m process variables $\{x_1, x_2, \ldots, x_m\}$ and one quality variable y. There are N available quality samples $\{y(t_1), y(t_2), \ldots, y(t_N)\}$ in total, where t_i $(1 \leq i \leq N)$ denotes the measurement time of the ith quality sample. For static soft sensor models like ordinary PLS, the ith input vector consists of merely m process samples that are measured at time t_i synchronized with the quality sample $y(t_i)$:

$$\mathbf{x}(i) = [x_1(t_i), x_2(t_i), \ldots, x_m(t_i)]^{\mathrm{T}}, \ 1 \leq i \leq N \tag{1}$$

As discussed in the previous section, such formulation provides no allowance for dynamic information in time series historical data. For the kth process variable $(1 \leq k \leq m)$, a historical input vector can be augmented by including lagged samples, which is described as:

$$\begin{aligned} \mathbf{x}_k(t_i) = &[x_k(t_i), x_k(t_i - \Delta t), \ldots, x_k(t_i - (n_f - 1)\Delta t)]^{\mathrm{T}} \\ &\in \mathbb{R}^{n_f}, \ 1 \leq i \leq N \end{aligned} \tag{2}$$

where Δt is the measurement interval for process variables and n_f is the length of historical input vectors. By stacking historical vector of m process variables into a column, we derive the input vector for SA models such as DPLS:

$$\mathbf{x}(i) = \begin{bmatrix} \mathbf{x}_1(t_i) \\ \mathbf{x}_2(t_i) \\ \vdots \\ \mathbf{x}_m(t_i) \end{bmatrix} \in \mathbb{R}^{mn_f},\ 1 \leq i \leq N \tag{3}$$

For simplicity, the time t_i is replaced by the index i in the following to enumerate the process samples $\{\mathbf{x}(i),\ 1 \leq i \leq N\}$.

2.2. Partial least squares (PLS)

In this study, the case of univariate output, i.e. PLS1, is considered. Given an input matrix $\mathbf{X} = [\mathbf{x}(1), \mathbf{x}(2), \ldots, \mathbf{x}(N)]^{\mathrm{T}} \in \mathbb{R}^{N \times m}$ and an output matrix $\mathbf{Y} = [y(t_1), y(t_2), \ldots, y(t_N)]^{\mathrm{T}} \in \mathbb{R}^{N}$, PLS projects input and output onto a low-dimensional subspace spread by A latent variables (LVs) $\{\mathbf{t}_1, \mathbf{t}_2, \cdots, \mathbf{t}_A\}$. Mathematically, the latent variable model is formed as:

$$\begin{aligned} \mathbf{X} &= \mathbf{T}\mathbf{P}^{\mathrm{T}} + \mathbf{E} \\ \mathbf{Y} &= \mathbf{T}\mathbf{Q}^{\mathrm{T}} + \mathbf{F} \end{aligned} \tag{4}$$

where $\mathbf{T} = [\mathbf{t}_1, \mathbf{t}_2, \cdots, \mathbf{t}_A] \in \mathbb{R}^{N \times A}$ denotes the score matrix, and $\mathbf{P} = [\mathbf{p}_1, \mathbf{p}_2, \cdots, \mathbf{p}_A] \in \mathbb{R}^{m \times A}$, $\mathbf{Q} = [\mathbf{q}_1, \mathbf{q}_2, \cdots, \mathbf{q}_A] \in \mathbb{R}^{1 \times A}$ are the loading matrices for $\mathbf{X}$ and $\mathbf{Y}$. Matrices $\mathbf{E}$ and $\mathbf{F}$ represent modeling residual of $\mathbf{X}$ and $\mathbf{Y}$. The objective embedded in PLS1 is to sequentially solve the following problem:

$$\begin{aligned} &\max_{\mathbf{w}_j}\ \mathbf{w}_j^{\mathrm{T}} \mathbf{X}_j^{\mathrm{T}} \mathbf{Y}_j \mathbf{Y}_j^{\mathrm{T}} \mathbf{X}_j \mathbf{w}_j \\ &\text{s.t. } \mathbf{w}_j^{\mathrm{T}} \mathbf{w}_j = 1 \end{aligned} \tag{5}$$

where $\mathbf{w}_j$ is the weight vector for the jth latent variable, which yields the eigenvector of $\mathbf{X}_j^{\mathrm{T}}\mathbf{Y}_j\mathbf{Y}_j^{\mathrm{T}}\mathbf{X}_j$ corresponding to the largest eigenvalue. The score vector is then derived as $\mathbf{t}_j = \mathbf{X}_j\mathbf{w}_j$. The loading vectors for $\mathbf{X}$ and $\mathbf{Y}$ are calculated as $\mathbf{p}_j = \mathbf{X}_j^{\mathrm{T}}\mathbf{t}_j/\mathbf{t}_j^{\mathrm{T}}\mathbf{t}_j$ and $\mathbf{q}_j = \mathbf{Y}_j^{\mathrm{T}}\mathbf{t}_j/\mathbf{t}_j^{\mathrm{T}}\mathbf{t}_j$. Then the jth latent variable is removed and input and output matrices for the $(j+1)$th latent variable are derived as $\mathbf{X}_{j+1} = \mathbf{X}_j - \mathbf{t}_j\mathbf{p}_j^{\mathrm{T}}$ and $\mathbf{Y}_{j+1} = \mathbf{Y}_j - \mathbf{t}_j\mathbf{q}_j^{\mathrm{T}}$. With A latent variables obtained, the regression equation can be finally written as [27]:

$$\mathbf{Y} = \mathbf{X}\mathbf{W}\left(\mathbf{P}^{\mathrm{T}}\mathbf{W}\right)^{-1}\mathbf{Q}^{\mathrm{T}} + \mathbf{F} \tag{6}$$

where $\mathbf{W} = [\mathbf{w}_1, \mathbf{w}_2, \cdots, \mathbf{w}_A]$. Given an out-of-sample data point $\mathbf{x}_{\mathrm{new}}$, the prediction model is expressed as:

$$\hat{y} = \beta^{\mathrm{T}}\mathbf{x}_{\mathrm{new}}. \tag{7}$$

where $\beta = \mathbf{W}\left(\mathbf{P}^{\mathrm{T}}\mathbf{W}\right)^{-1}\mathbf{Q}^{\mathrm{T}}$.

3. Dynamic partial least squares with temporal smoothness

Data modeling in a dynamic environment is a common task in many practical applications. With regard to this topic, the temporal smoothness has been assumed in data-driven models in different areas recently [28], [29]. In this section, we inspire the idea of the temporal smoothness regularization first within the framework of the ordinary DPLS. Then we present the soft sensor modeling approach based DPLS with temporal smoothness regularizations.

3.1. Problem formulation

As a latent variable model, PLS uses LVs $\{\mathbf{t}_1, \mathbf{t}_2, \cdots, \mathbf{t}_A\}$ to explain most variances in $\mathbf{X}$ and $\mathbf{Y}$ spaces. The score variable $\mathbf{t}_j$ of central focus is obtained by projecting input matrix $\mathbf{X}$ onto the direction $\mathbf{w}_j$. Here the input sample vector with historical variables defined in (3) instead of the usual (1) is adopted. Similarly, the corresponding direction vector $\mathbf{w}_j$ can be decomposed as m coefficient vectors defined as follows:

$$\mathbf{w}_j = \begin{bmatrix} \mathbf{w}_{j,1} \\ \mathbf{w}_{j,2} \\ \vdots \\ \mathbf{w}_{j,m} \end{bmatrix} \in \mathbb{R}^{mn_f} \tag{8}$$

where $\mathbf{w}_{j,k} = [w_{j,k}(1), w_{j,k}(2), \ldots, w_{j,k}(n_f)]^{\mathrm{T}}$ is the coefficient vector of $\mathbf{x}_k(t_i)$. In PLS, low-dimensional features underlying high-dimensional input vectors are represented by LVs $\mathbf{t}_j$, of which the ith element is calculated as:

$$\begin{aligned} & t_j(i) \\ = & \mathbf{x}(i)^{\mathrm{T}} \mathbf{w}_j \\ = & \sum_{k=1}^{m} \mathbf{x}_k(t_i)^{\mathrm{T}} \mathbf{w}_{j,k} \\ = & \sum_{k=1}^{m} \sum_{l=1}^{n_f} x_k(t_i - (l-1)\Delta t) w_{j,k}(l). \end{aligned} \tag{9}$$

Intuitively, each process variable with lagged samples can be conceived to contribute to the LV $\mathbf{t}_j$ by convolution with a coefficient vector $\mathbf{w}_{j,k}$, and each lagged sample has its own coefficient $w_{j,k}(l)$ in (9). In chemical processes, successively fast-sampled data should have smoothly varying impacts on inherent features $\{\mathbf{t}_j\}$. In other words, the coefficients of proximal historical samples $x_k(t_i - l\Delta t)$ and $x_k(t_i - (l-1)\Delta t)$, should be temporally similar. Therefore, in order to encourage temporal smoothness in weighed coefficients, we prefer a minor difference between proximal coefficients $w_{j,k}(l+1)$ and $w_{j,k}(l)$. A pervasive choice for the penalty term is the L_2 regularization:

$$\sum_{l=1}^{n_f - 1} [w_{j,k}(l+1) - w_{j,k}(l)]^2 \tag{10}$$

which can be termed as the *temporal smoothness regularization*. Then minimization of (10) can be neatly re-written as:

$$\min_{\mathbf{w}_{j,k}} \mathbf{w}_{j,k}^{\mathrm{T}} \mathbf{J}^{\mathrm{T}} \mathbf{J} \mathbf{w}_{j,k} \tag{11}$$

where

$$\mathbf{J} = \begin{bmatrix} 1 & -1 & & \\ & \ddots & \ddots & \\ & & 1 & -1 \end{bmatrix} \in \mathbb{R}^{(n_f-1)\times n_f}. \tag{12}$$

Considering coefficient vectors $\mathbf{w}_j = \left[\mathbf{w}_{j,1}^{\mathrm{T}}, \mathbf{w}_{j,2}^{\mathrm{T}}, \ldots, \mathbf{w}_{j,m}^{\mathrm{T}}\right]^{\mathrm{T}}$ of all m process variables, the smoothness penalty for $\mathbf{w}_j$ is derived as

$$\min_{\mathbf{w}_j} \mathbf{w}_j^{\mathrm{T}} \mathbf{K} \mathbf{w}_j \tag{13}$$

where $\mathbf{K} = \mathbf{I}_m \otimes \mathbf{J}^{\mathrm{T}}\mathbf{J}$, and $\otimes$ denotes the Kronecker product. Because $\mathbf{w}_j$ is calculated by solving an eigenvector problem, we combine the L_2 penalty term with the objective in the optimization problem in (5), expressed as follows:

$$\begin{aligned} &\max_{\mathbf{w}_j} \ \mathbf{w}_j^{\mathrm{T}} \mathbf{X}_j^{\mathrm{T}} \mathbf{Y}_j \mathbf{Y}_j^{\mathrm{T}} \mathbf{X}_j \mathbf{w}_j - \alpha \mathbf{w}_j^{\mathrm{T}} \mathbf{K} \mathbf{w}_j \\ &\ \text{s.t. } \mathbf{w}_j^{\mathrm{T}} \mathbf{w}_j = 1 \end{aligned} \tag{14}$$

where $\alpha \geq 0$ denotes the regularization parameter. The first term in the objective pursues a direction that maximizes the covariance between $\mathbf{X}_j \mathbf{w}_j$ and $\mathbf{Y}_j$, whereas the second term aims at enhancing smoothness of coefficients. The optimization problem in (14) remains an simple eigenvector problem, whose solution is the eigenvector corresponding to the largest eigenvalue. This formulation is therefore computationally handy in practice.

Table 1: Training algorithm for DPLS with temporal smoothness

Set $j = 1$ and $\mathbf{X}_j = \mathbf{X},\ \mathbf{Y}_j = \mathbf{Y}$.
1. Calculate $\mathbf{w}_j$ as the eigenvector of $\mathbf{X}_j^{\mathrm{T}}\mathbf{Y}_j\mathbf{Y}_j^{\mathrm{T}}\mathbf{X}_j - \alpha\|\|\mathbf{X}_j^{\mathrm{T}}\mathbf{Y}_j\|\|^2\mathbf{K}$ corresponding to the largest eigenvalue.
2. $\mathbf{t}_j = \mathbf{X}_j\mathbf{w}_j$.
3. $\mathbf{p}_j = \mathbf{X}_j^{\mathrm{T}}\mathbf{t}_j/\mathbf{t}_j^{\mathrm{T}}\mathbf{t}_j$ and $\mathbf{q}_j = \mathbf{Y}_j^{\mathrm{T}}\mathbf{t}_j/\mathbf{t}_j^{\mathrm{T}}\mathbf{t}_j$.
4. $\mathbf{X}_{j+1} = \mathbf{X}_j - \mathbf{t}_j\mathbf{p}_j^{\mathrm{T}}$ and $\mathbf{Y}_{j+1} = \mathbf{Y}_j - \mathbf{t}_j\mathbf{q}_j^{\mathrm{T}}$.
Set $j = j + 1$ and return to step 1. Terminate if $j > A$.

However, the optimization scheme devised in (14) might have some limitations. By scrutinizing (5), we can find that the objective function remains unchanged if we scale the input or output matrix by a constant coefficient. For example, when $\mathbf{X}_j$ and $\mathbf{Y}_j$ are replaced by $a\mathbf{X}_j$ and $b\mathbf{Y}_j$ where $\{a, b\}$ are non-zero constants, the objective in (14) is still identical to maximizing $\mathbf{w}_j^{\mathrm{T}}\mathbf{X}_j^{\mathrm{T}}\mathbf{Y}_j\mathbf{Y}_j^{\mathrm{T}}\mathbf{X}_j\mathbf{w}_j$. This is a reasonable and important asset because correlation relationship ought to be irrelevant to the scale of $\mathbf{X}_j$ and $\mathbf{Y}_j$. However, it is obvious that the problem with temporal smoothness regularization possesses no invariance given a non-zero α. In this regard, (14) is modified as follows:

$$\begin{aligned} &\max_{\mathbf{w}_j} \mathbf{w}_j^{\mathrm{T}} \left(\mathbf{X}_j^{\mathrm{T}}\mathbf{Y}_j\mathbf{Y}_j^{\mathrm{T}}\mathbf{X}_j\mathbf{w}_j - \alpha\|\|\mathbf{X}_j^{\mathrm{T}}\mathbf{Y}_j\|\|^2\mathbf{K} \right) \mathbf{w}_j \\ &\ \text{s.t. } \mathbf{w}_j^{\mathrm{T}}\mathbf{w}_j = 1 \end{aligned} \tag{15}$$

In this way the projection vector $\mathbf{w}_j$ becomes invariant to the linear scaling of input and output matrices. At last, the above formulation in (15) is adopted to calculate $\mathbf{w}_j$. The rest steps in establishing a DPLS model are identical to those in ordinary PLS aforementioned. The entire procedure of the univariate DPLS-TS algorithm is listed in Table 1. Hyper-parameters such as the regularization parameter α and the number of selected LVs A can be determined using cross-validation in this context.

3.2. Soft sensor development based on DPLS with temporal smoothness

In summary, the procedure of soft sensor modeling based on DPLS with temporal smoothness includes the following steps:

Step 1: Select proper process variables and the length of historical data n_f according to prior process knowledge.

Step 2: Remove potential outliers from original data with some given criterion. Then normalize the data such that all samples of a certain process variable has zero mean and unit variance.

Step 3: Set a grid in the space of $\{A, \alpha\}$, and the number v of folds in cross-validation.

Step 4: For each pair of $\{A, \alpha\}$, train a dynamic soft sensor model v times on different training datasets using the algorithm given in Table 1, and then calculate the cross-validation RMSE.

Step 5: Choose the $\{A, \alpha\}$ with least cross-validation root mean square error (RMSE) among all nodes in the grid, which is defined as follows:

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{N} ||y(i) - \hat{y}(i)||^2}{N}}. \tag{16}$$

4. Case study on simulation examples

4.1. A numerical example

In this subsection, a numerical example is provided to illustrate the advantages of dynamic models over static counterparts, and further addresses the benefit brought by temporal smoothness regularizations. The design of the experimental dataset is motivated by Ref. [30], [31]. The number of process variables and the length of historical data are set as $m = 4$ and $n_f = 12$ respectively. The system is formulated as:

$$y = \beta^{\mathrm{T}}\mathbf{x} + \epsilon = \sum_{k=1}^{4} \beta_k^{\mathrm{T}}\mathbf{x}_k + \epsilon. \tag{17}$$

The input vector $\mathbf{x}$ is augmented by stacking historical data vector as $\mathbf{x} = \left[\mathbf{x}_1^{\mathrm{T}}\ \mathbf{x}_2^{\mathrm{T}}\ \mathbf{x}_3^{\mathrm{T}}\ \mathbf{x}_4^{\mathrm{T}}\right]^{\mathrm{T}}$. $\mathbf{x}$ is assumed to take a multivariate normal distribution $\mathbf{x} \sim N(\mathbf{0}, \Omega^{-1})$ with zero mean and covariance matrix Ω, in which the entry is set as:

$$\Omega_{n_f(k-1)+l, n_f(k'-1)+l'} = 0.95^{|k-k'|} \exp(-|l-l'|), \tag{18}$$

In comparison with the example in [30], this definition assumes additional correlations between historical samples. The regression parameter vector β can be decomposed as $\left[\beta_1^{\mathrm{T}}\ \beta_2^{\mathrm{T}}\ \beta_3^{\mathrm{T}}\ \beta_4^{\mathrm{T}}\right]^{\mathrm{T}}$, where $\beta_k \in \mathbb{R}^{12}$ denotes the regression parameter vector for $\mathbf{x}_k$. To describe dynamics with respect to process variables, β_k is assumed to be the FIR of a low-order transfer function $G_k(s)$ with unit gain in the following form:

$$G_k(s) = \frac{1}{T_1 s^2 + T_2 s + 1}\ (1 \le k \le 4). \tag{19}$$

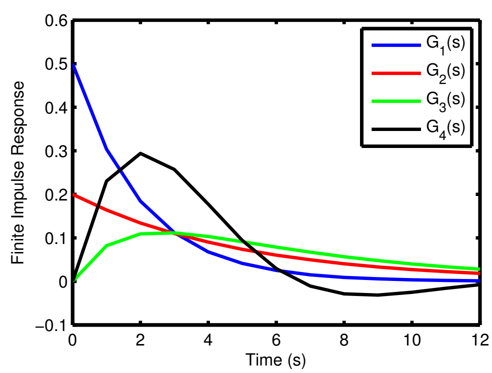

Figure 5: Finite impulse responses of transfer functions in the numerical example

Parameters $\{T_1, T_2\}$ in transfer functions $G_k(s)$ are tabulated in Table 2. According to the low-order characteristic of chemical processes, two tranfer functions are set as second-order, while the other two are set as first-order. The sampling interval for FIRs is set as 1 second. Fig. 5 displays FIRs of $G_k(s)$, from which we can clearly see the settling time for this numerical example is

Table 2: Parameters of Transfer Functions in the Numerical Example

	$G_1(s)$	$G_2(s)$	$G_3(s)$	$G_4(s)$
T_1	0	0	8	3
T_2	2	5	7	2

about 12s. ϵ in (17) denotes the sampling noise in quality data, which is Gaussian distributed as $\epsilon \sim N(0, \sigma_y^2)$. The noise variance σ_y^2 is set as:

$$\sigma_y = 0.3 \times \sqrt{\text{var}\left(\vec{\beta}^{\mathrm{T}}\mathbf{x}\right)}. \tag{20}$$

Three approaches, namely PLS, DPLS and DPLS-TS, are applied to train the soft sensor models. In this study, the number of PCs and the regularization parameter α are determined using 10-fold cross-validation. For a fair comparison, the length n_f of historical vector is set as 12 in both DPLS and DPLS-TS according to prior knowledge. In order to show the efficiency of the comparison, 50 Monte Carlo simulations are performed to generate input and output data from the given multivariate Gaussian distribution. In each simulation, 150 samples are generated as training data and 150 samples as test data. Then the training procedure for PLS, DPLS and DPLS-TS is repeated 50 times for each training set, and the modeling statistics are obtained, which are reported in Table 3. As expected, the static PLS has a much poorer prediction accuracy than dynamic models in term of prediction RMSEs when there exists evident dynamics. In this regard dynamic soft sensors are preferred. DPLS-TS has less RMSE than ordinary PLS, which illustrates the power of proposed temporal smoothness regularizations. Next, we use the quadratic loss $\hat{\beta}^{\mathrm{T}}K\hat{\beta}$ to quantify the smoothness of model parameters, where $\hat{\beta}$ is the regressor derived in DPLS-TS and DPLS and K is defined in (13). From the second row in Table 3, it is perspicuous that the proposed model has improved smoothness in dynamic parameters, which is the purpose of DPLS-TS. The third row gives average results of $||\hat{\beta} - \beta||_2$, which evaluates the discrepancy between the true value β and the estimated one $\hat{\beta}$ thereof. It is observed that model parameters are better

Table 3: Simulation Results in the Numerical Example (mean values in 50 Monte Carlo simulations)

	DPLS-TS	DPLS	PLS
Prediction RMSE	**0.1847**	0.2199	1.2328
Smoothness Loss	**0.0243**	0.0438	NaN
$\|\|\hat{\beta} - \beta\|\|_2$	**0.2988**	0.3064	NaN

recognized by the proposed method.

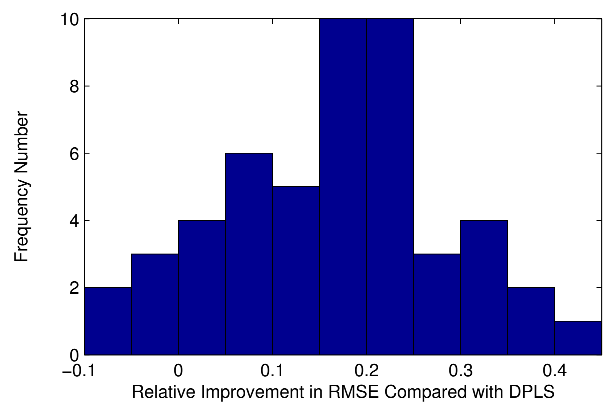

Figure 6: Relative improvement of DPLS-TS in RMSE values in comparison with DPLS in 50 Monte Carlo simulations.

Fig. 6 further presents the improvements in prediction RMSE of DPLS-TS compared with DPLS in 50 simulations. The relative improvement is defined as $1 - \text{RMSE}_{\text{DPLS-TS}}/\text{RMSE}_{\text{DPLS}}$. It is observed that DPLS-TS has better prediction accuracy in 90 percent of all cases, and there are 70 percent of cases in which the relative improvement in RMSE is larger than 0.1. From the results in Table 3 and Fig. 6, we can see that the proposed method can desirably improve the generalization of dynamic soft sensor models.

Next we make some discussions of the influence of model parameters on prediction performances. There are two hyper-parameters which are closely related to model dynamics, namely the regularization parameter α and the length of historical vector n_f. Fig. 7 gives the average RMSE curves with

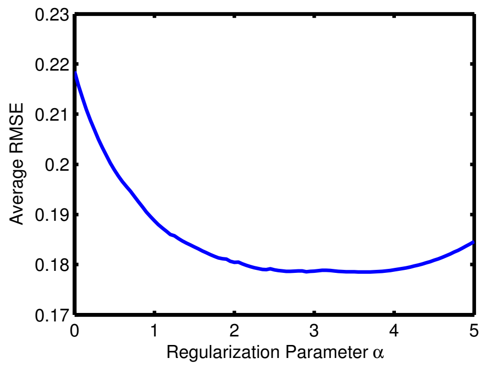

Figure 7: Average RMSE curve with different regularization parameter α in DPLS-TS.

respect to different regularization parameter α in DPLS-TS. Here the number of PCs is uniformly set as 3 for a fair comparison. Notice that when $\alpha = 0$, DPLS-TS reduces to the ordinary DPLS. The prediction performance gets notably improved with α increasing from zero, and then becomes optimal around 3, which corresponds with the result obtained by cross-validation. The prediction accuracy starts degrading with a over large α hereafter.

Finally, we examine the influence of the historical data length n_f on the estimation performance of two dynamic models. Table 4 lists the average RMSE values of different choices of n_f for DPLS and DPLS-TS in the above 50 Monte Carlo simulations. In general, in the presence of evident process dynamics, the prediction accuracies could be improved with more lagged data included. Notice that the case with $n_f = 1$ in DPLS reduces to the ordinary PLS. When n_f begins to increase from zero, it is observed that RMSEs of both models are reduced because of the historical information contained. The performances of two approaches are comparable when n_f is small, mainly because the historical variables that are used commonly have significant impacts on the output. Consequently, parameters are learnt mainly from the data and the regularization term is not as necessary as expected. However, when n_f continues to grow, the gap in two RMSEs grows evidently, in the sense that DPLS-TS tends to outperform DPLS. It is noted that DPLS reaches its minimal RMSE when $n_f = 10$.

Table 4: Average RMSE of Dynamic Soft Sensors under Different Historical Vector Length n_f

n_f	DPLS-TS	DPLS
1	NaN	1.2328
2	1.0422	1.0387
3	0.7368	0.7380
4	0.5073	0.5075
5	0.3539	0.3568
6	0.2581	0.2688
7	0.2205	0.2340
8	0.2043	0.2222
9	0.1973	0.2168
10	0.1902	**0.2154**
11	0.1859	0.2169
12	**0.1847**	0.2199

It overfits the data when $n_f > 10$ with an increase in RMSE values, because the model to be learnt becomes more intricated but the number of available training samples remains the same. In contrast, the performance of DPLS-TS is enhanced all through, being best when $n_f = 12$. A temporal smoothness regularization term is shown to help to alleviate over-parameterization and utilize more historical data effectively. Moreover, such a merit brings some practical benefits. It is common that the historical length n_f is selected by process practitioners according to a comprehensive prior knowledge, such as the response time of a certain process. However, due to the complexity of industrial processes, such prior knowledge might not be obtained accurately. A rough estimation of n_f may lead to over-fitting problem in ordinary DPLS, which is desirably mitigated by using temporal smoothness regularizations.

4.2. Tennessee Eastman benchmark process

The Tennessee Eastman (TE) process proposed by Downs and Vogel [32] has been a popular benchmark in a variety of process control tasks, including model predictive control (MPC), soft sensor design, process monitoring and so forth. The decentralized control strategy proposed in [33] is applied in this study. The TE process has 12 manipulated variables XMV(1-12) and 41 measured variables XMEAS(1-41). In this study, XMEAS(1-22), XMV(1-4), (6-8) and (10,11) are chosen as process variables. Variables XMV(5), (9) and (12) have been excluded because they keep constant values in the control strategy applied here. The sampling interval for process data is set as 3 minutes, and the length of historical data is set as $n_f = 5$. XMEAS(30), namely the composition of B in Stream 9 is chosen as the quality variable in this context, whose sampling interval is 6 minutes. Overall, 1440 samples under the normal condition are produced, while 480 samples are training data and the rest 960 samples are test data.

The DPLS and DPLS-TS are applied to construct the soft sensor model. Hyper-parameters such as the number of LVs and the regularization parameter are determined using 10-fold cross-validation. The optimal hyper-parameters of DPLS-TS are chosen as $A = 4$ and $\alpha = 10$ with a cross-validation RMSE of 0.1122, while for DPLS $A = 2$ with a cross-validation RMSE of 0.1137. Cross-validation procedures reveal surprisingly that an optimal structure of DPLS-TS is more intricated than that of DPLS, since four LVs are selected in DPLS-TS but only two in DPLS. Table 5 further gives prediction results of different approaches, in which we observe that DPLS-TS outperforms DPLS not only in cross-validations, but also in the test dataset. Comparison with both cross-validation and test errors demonstrate that, even if a more complicated structure is achieved by DPLS-TS, it is better than DPLS yet. This is because the proposed model better agrees with the physical truth of processes in terms of dynamic behaviors. It is convincing that the temporal smoothness regularization is able to extract more useful features from process data effectively, and helps to find an appropriate model structure.

Table 5: Modeling Results in Tennessee Eastman Benchmark Process

	DPLS-TS	DPLS
Test RMSE	0.1083	0.1160
Smoothness Loss	0.0003	0.0353

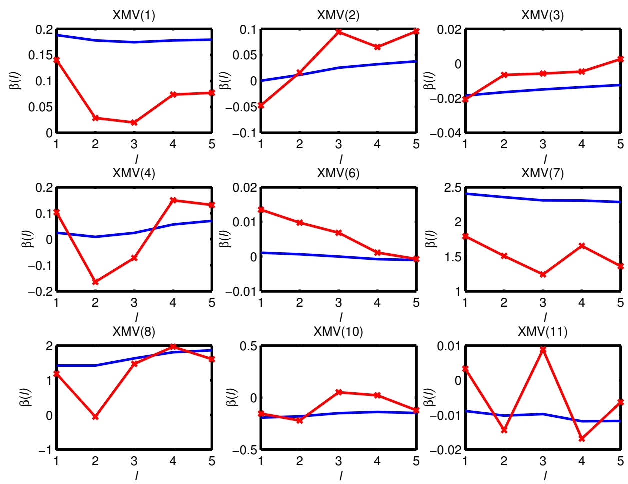

Figure 8: Regression coefficients of manipulated variables (XMV) estimated by DPLS-TS and DPLS. DPLS: the red line with markers; DPLS-TS: the blue line without markers.

From the second row of Table 5, it can be seen that DPLS-TS has less smoothness loss in regression parameters due to the temporal smoothness regularization. Fig. 8 and Fig. 9 intuitively visualize the estimated regression coefficients of manipulated variables and measurement variables respectively. At first glance, the regressor vectors obtained by two methods for one process variable are somewhat different in amplitudes thereof. This is due to the fact that they have different number of LVs and thus yield different model structures. Therefore the focus here is simply on the temporal smoothness. We can observe that all regression coefficients are well smoothed by DPLS-TS, better revealing the trend of process dynamics and being interpretable. As a matter of fact, severe variations in dynamic behavior seem unlikely to occur when the process

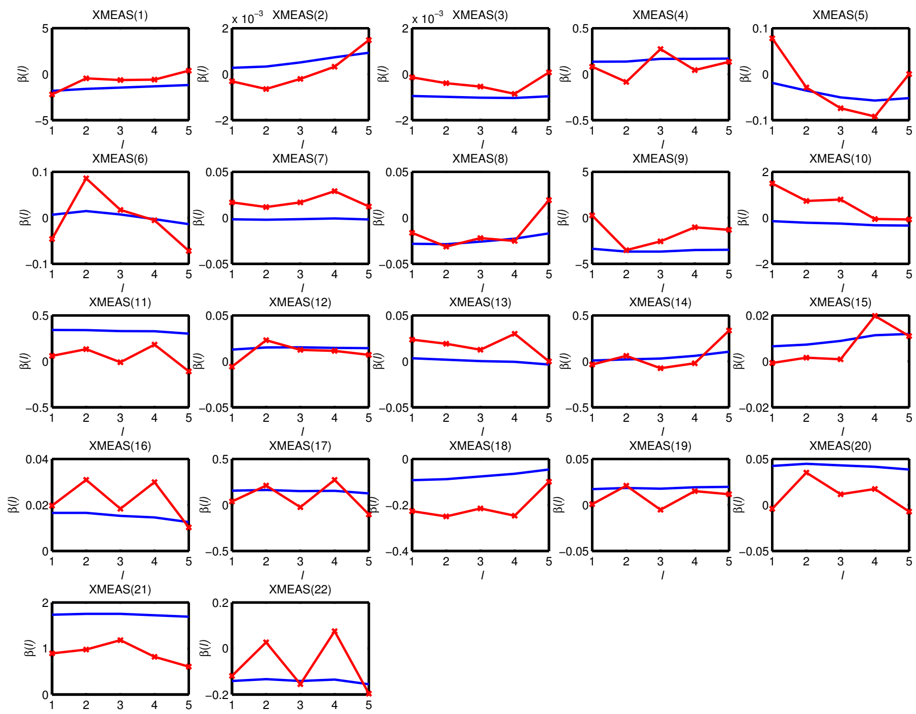

Figure 9: Regression coefficients of measured variables (XMEAS) estimated by DPLS-TS and DPLS. DPLS: the red line with markers; DPLS-TS: the blue line without markers.

is operated in a normal condition, while some coefficients obtained by DPLS oscillate dramatically, for example, those of XMV(4), XMV(11), XMEAS(16) and XMEAS(22). Oscillating coefficients are satisfactorily avoided by using a temporal smoothness regularizations. In addition, smoothed dynamic coefficients are particularly beneficial in design of proper quality control schemes such as MPC, which deserves further investigation.

5. Industrial application on crude distillation unit

The crude distillation unit (CDU) is a core part in petrochemical industry. In the CDU, crude is partitioned into different fractions, and a variety of products are obtained such as naphtha, kerosene, light diesel and heavy diesel. To improve the yield of products and keep the plant operation safe, real-time control of product quality indices such as boiling/ash/pour points is of great importance. However, these indices are often measured via laboratory analysis, which takes several hours to accomplish and involves extensive manual workloads. Consequently, for quality control purposes, the requirement of real-time estimations cannot be satisfied. In practice, soft sensing techniques is commonly utilized to provide online estimations of quality indices.

Here soft sensors are established to predict the ASTM (American Society for Testing Materials) 95% cut point of heavy diesel. All data come from real measurements and records of a refinery plant in northwest China. In total, 20 input variables have been selected, which are reported in Table 6. There are 453 quality samples archived through one year's laboratory analysis. The dataset is randomly partitioned into a training set (226 samples) and a test set (227 samples). The sampling interval for process variables such as temperatures, pressures, flows and liquid levels is two minutes. To describe process dynamics, the length of historical data is chosen as 12, and both DPLS-TS and DPLS models are developed for comparison in this study. The optimal number of LVs in both approaches is selected as 4 via cross-validation, whereas the regularization parameter α in DPLS-TS is selected as 25.

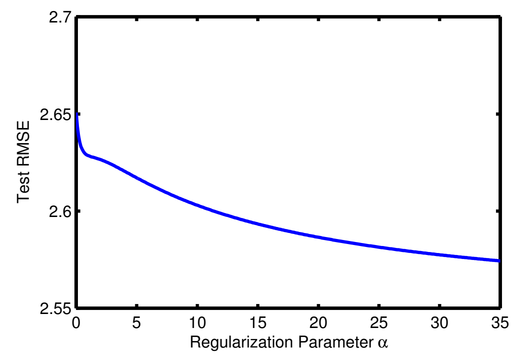

Figure 10: RMSE value on the test data with different regularization parameter α. The ordinary DPLS is a special case of DPLS-TS with $\alpha = 0$.

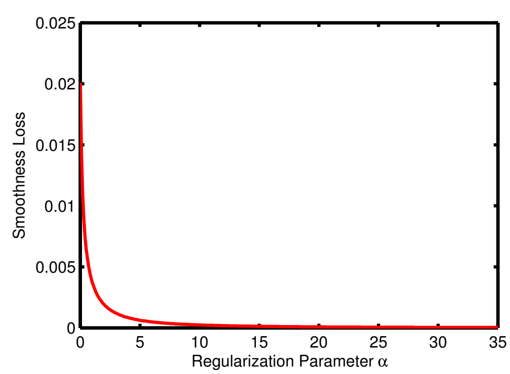

Figure 11: Smoothness loss of regression coefficients with different regularization parameter α. The ordinary DPLS is a special case of DPLS-TS with $\alpha = 0$.

Table 6: Process Variables in the Crude Distillation Unit

No.	Description
1	Top temperature
2	Top pressure
3	Reflux flow
4	Side-drawn 1# tray temperature
5	Side-drawn 1# flow rate ratio
6	Side-drawn 2# tray temperature
7	Side-drawn 2# flow rate ratio
8	Side-drawn 3# tray temperature
9	Side-drawn 3# flow rate ratio
10	Steam flow rate ratio
11	Top recycle heat
12	Middle recycle 1# heat
13	Middle recycle 2# heat
14	Top recycle drawn temperature
15	Feed temperature
16	Top flow rate
17	Upper-drawn 4# tray temperature
18	Sub-drawn 4# tray temperature
19	Reflux flow ratio from 4# tray
20	Blending ratio of two different crude feeds

Detailed modeling results of DPLS-TS and DPLS are presented below. Because DPLS corresponds to the special case of DPLS-TS with $\alpha = 0$, we are able to directly compare test RMSEs and smoothness losses with different regularization parameter α of DPLS-TS, as shown in Fig. 10 and 11 respectively. In Fig. 10, we observe that the ordinary DPLS with $\alpha = 0$ has the worst prediction accuracy. When α deviates from zero within a small range, the test RMSE value drops rather significantly. With α continuing to increase, the prediction performance gets better improved thanks to temporal smoothness regularizations. Fig. 11 plots the smoothness losses with different α, which shows a similar trend to Fig. 10. The ordinary DPLS with $\alpha = 0$ has the largest smoothness loss. By contrast, when $\alpha > 0$, the smoothness loss is reduced dramatically, indicating that the regularization term takes effect and the model becomes more interpretable. From Fig. 10 and 11, it can be seen that DPLS-TS in general achieves both improved prediction accuracy and smoothness in regression coefficients. In addition, we observe that prediction accuracy is closely correlated with temporal smoothness. This is because the dynamic nature of processes is better described by temporal smoothness regularizations of merit and the interpretability of models is enhanced.

6. Conclusions and perspectives

In this article, a DPLS based soft sensor modeling approach with temporal smoothness has been proposed. Different from the ordinary DPLS, the proposed model penalizes significant changes in dynamic parameters. With a temporal smoothness regularization introduced in the derivation of LVs, the model agrees better with the physical truth of chemical processes. Compared to the ordinary DPLS model, the proposed approach has better interpretability and yields improved generalizations, particularly when the length of historical data is large or there exist evident dynamics in the process. The optimization problem induced by the temporal smoothness regularization takes the form of an eigenvector problem that are computationally efficient. Two simulated examples and an industrial data case study have indicated the efficacy of the proposed model.

This study merely focuses on linear models when static soft sensors are directly generalized to time series historical data. When nonlinear models like neural networks are extended to dynamic versions as such, more considerable over-fitting concern, however, would be encountered because of more parameters to be optimized and more complicated architectures than their linear counterparts. Nonlinear parameters for various lagged variables would interact excessively with each other so that the model tends to over-accommodate the data. In this sense the temporal smoothness is necessary for improving the interpretations as well as generalizations of nonlinear soft sensor models, which is worth studying in the future.

ACKNOWLEDGMENT

This work is supported in part by the National Basic Research Program of China (2012CB720505), Tsinghua University Initiative Scientific Research Program and BIL Project with KU Leuven. Xiaolin Huang and Johan Suykens acknowledge support from KU Leuven, the Flemish government, FWO, the Belgian federal science policy office and the European Research Council (CoE EF/05/006, GOA MANET, IUAP DYSCO, FWO G.0377.12, BIL Project with Tsinghua University, ERC AdG A-DATADRIVE-B). The scientific responsibility is assumed by its authors.

References

[1] M. Kano, M. Ogawa, "The state of the art in chemical process control in Japan: Good practice and questionnaire survey," *Journal of Process Control*, vol. 20, no. 9, pp. 969–982, 2010.

[2] F. Yacoub and J. F. MacGregor, "Robust processes through latent variable modeling and optimization," *AIChE Journal*, vol. 57, no. 5, pp. 1278–1287, 2011.

[3] J. Mori, J. Yu, "Quality relevant nonlinear batch process performance monitoring using a kernel based multiway non-Gaussian latent subspace projection approach," *Journal of Process Control*, vol. 24, no. 1, pp. 57–71, 2014.

[4] P. Kadlec, B. Gabrys, and S. Strandt, "Data-driven soft sensors in the process industry," *Computers & Chemical Engineering*, vol. 33, no. 4, pp. 795–814, 2009.

[5] S. Joe Qin, "Recursive PLS algorithms for adaptive data modeling," *Computers & Chemical Engineering*, vol. 22, no. 4, pp. 503–514, 1998.

[6] P. Facco, F. Doplicher, F. Bezzo, and M. Barolo, "Moving average PLS soft sensor for online product quality estimation in an industrial batch polymerization process," *Journal of Process Control*, vol. 19, no. 3, pp. 520–529, 2009.

[7] H. Kaneko, M. Arakawa, and K. Funatsu, "Development of a new soft sensor method using independent component analysis and partial least squares," *AIChE Journal*, vol. 55, no. 1, pp. 87–98, 2009.

[8] S. Joe Qin, "Neural networks for intelligent sensors and control—," *Neural Systems for Control*, p. 213, 1997.

[9] C. Shang, F. Yang, D. Huang, and W. Lyu, "Data-driven soft sensor development based on deep learning technique," *Journal of Process Control*, vol. 24, no. 3, pp. 223–233, 2014.

[10] W. Yan, H. Shao, and X. Wang, "Soft sensing modeling based on support vector machine and Bayesian model selection," *Computers & Chemical Engineering*, vol. 28, no. 8, pp. 1489–1498, 2004.

[11] K. Desai, Y. Badhe, S. S. Tambe, and B. D. Kulkarni, "Soft-sensor development for fed-batch bioreactors using support vector regression," *Biochemical Engineering Journal*, vol. 27, no. 3, pp. 225–239, 2006.

[12] S. Khatibisepehr and B. Huang, "Dealing with irregular data in soft sensors: Bayesian method and comparative study," *Industrial & Engineering Chemistry Research*, vol. 47, no. 22, pp. 8713–8723, 2008.

[13] M. Kano, K. Miyazaki, S. Hasebe, and I. Hashimoto, "Inferential control system of distillation compositions using dynamic partial least squares regression," *Journal of Process Control*, vol. 10, no. 2, pp. 157–166, 2000.

[14] P. Cao and X. Luo, "Modeling of soft sensor for chemical process," *CIESC Journal*, vol. 3, p. 004, 2013.

[15] N. Bhat and T. J. McAvoy, "Use of neural nets for dynamic modeling and control of chemical process systems," *Computers & Chemical Engineering*, vol. 14, no. 4, pp. 573–582, 1990.

[16] X. Gao, F. Yang, D. Huang, and Y. Ding, "An iterative two-level optimization method for the modeling of Wiener structure nonlinear dynamic soft sensors," *Industrial & Engineering Chemistry Research*, vol. 53, no. 3, pp. 1172–1178, 2014.

[17] P. Cao and X. Luo, "Modeling for soft sensor systems and parameters updating online," *Journal of Process Control*, vol. 24, no. 6, pp. 975–990, 2014.

[18] Y. Ma, D. Huang, and Y. Jin, "Discussion about dynamic soft-sensing modeling," *Journal of Chemical Industry and Engineering (China)*, vol. 56, no. 8, p. 1516, 2005.

[19] Y. Wu and X. Luo, "A novel calibration approach of soft sensor based on multirate data fusion technology," *Journal of Process Control*, vol. 20, no. 10, pp. 1252–1260, 2010.

[20] C. Shang, X. Gao, F. Yang, and D. Huang, "Novel Bayesian framework for dynamic soft sensor based on support vector machine with finite impulse response," *IEEE Transactions on Control Systems Technology*, vol. 22, no. 4, pp. 1550–1557, 2014.

[21] B. Lin, B. Recke, J. K. Knudsen, and S. B. Jørgensen, "A systematic approach for soft sensor development," *Computers & Chemical Engineering*, vol. 31, no. 5, pp. 419–425, 2007.

[22] H. J. Galicia, Q. P. He, and J. Wang, "A reduced order soft sensor approach and its application to a continuous digester," *Journal of Process Control*, vol. 21, no. 4, pp. 489–500, 2011.

[23] I. S. Helland, "Some theoretical aspects of partial least squares regression," *Chemometrics and Intelligent Laboratory Systems*, vol. 58, no. 2, pp. 97–107, 2001.

[24] V. N. Vapnik, *Statistical Learning Theory.* Wiley New York, 1998, vol. 2.

[25] J.A.K. Suykens, T. Van Gestel, J. De Brabanter, B. De Moor, and J. Vandewalle, *Least Squares Support Vector Machines.* World Scientific, Singapore, 2002.

[26] R. G. Baraniuk, "Compressive sensing," *IEEE Signal Processing Magazine*, vol. 24, no. 4, 2007.

[27] B. Dayal, J. F. MacGregor, *et al.*, "Improved PLS algorithms," *Journal of Chemometrics*, vol. 11, no. 1, pp. 73–85, 1997.

[28] H. Ohlsson, L. Ljung, S. Boyd, "Segmentation of ARX-models using sum-of-norms regularization," *Automatica*, vol. 46, no. 6, pp. 1107-1111, 2010.

[29] R. Langone, C. Alzate, J.A.K. Suykens , "Kernel spectral clustering with memory effect," *Physica A*, vol. 392, no. 10, pp. 2588–2606, 2013.

[30] I.-G. Chong and C.-H. Jun, "Performance of some variable selection methods when multicollinearity is present," *Chemometrics and Intelligent Laboratory Systems*, vol. 78, no. 1, pp. 103–112, 2005.

[31] M. Hebiri, S. van de Geer, *et al.*, "The smooth-lasso and other 1+ 2-penalized methods," *Electronic Journal of Statistics*, vol. 5, pp. 1184–1226, 2011.

[32] J. J. Downs and E. F. Vogel, "A plant-wide industrial process control problem," *Computers & Chemical Engineering*, vol. 17, no. 3, pp. 245–255, 1993.

[33] N. Lawrence Ricker, "Decentralized control of the Tennessee Eastman challenge process," *Journal of Process Control*, vol. 6, no. 4, pp. 205–221, 1996.